

Intel® C++ Compiler 6.0 for Linux*

Getting Started Guide

Contents

1. [Overview](#)
2. [Supported Linux Distributions](#)
3. [Installation Notes](#)
4. [Building "Hello World"](#)
5. [Getting Started with Compiler Optimizations](#)
6. [Additional Information](#)
7. [System Requirements](#)

Overview

This document explains how to install the Intel® C++ compiler for Linux*, build "Hello World" for IA-32 and Intel® Itanium™ processor based systems and how to get started optimizing your applications with the Intel compilers.

The Intel C++ Compiler 6.0 for Linux consists of the following:

- Intel C++ Compiler for IA-32 based applications: `icc`
- Intel C++ Compiler for Itanium processor-based applications: `ecc`
- Linux Application Debugger 6.0 for IA-32 based applications: `ldb`
- Linux Application Debugger 6.0 for Itanium based applications: `ldb`
- Intel Itanium Assembler 6.0 for Itanium based applications: `ias`
- Product documentation

Supported Linux Distributions

This version of the Intel compilers for Linux supports Linux distributions with the following combinations of glibc, the C language library, and the Linux kernel.

Supported IA-32 based systems:

- Distribution with 2.2.2 glibc & 2.4 kernel; or
- Distribution with 2.2.4 glibc & 2.4 kernel.

Supported Itanium processor based systems:

- Distribution with 2.2.3 glibc & 2.4 kernel; or
- Distribution with 2.2.4 glibc & 2.4 kernel.

System Requirements

IA-32 Processor System Requirements

- A computer based on a Pentium processor or subsequent IA-32 based processor, a Pentium 4 processor is recommended.
- 128 MB of RAM, 256 MB recommended.
- 100 MB of disk space
- A supported Linux distribution, see [Supported Linux Distributions](#) section.

Itanium Processor System Requirements

- A computer with an Itanium processor.
- 256 MB of RAM
- 100 MB of disk space
- A supported Linux distribution, see [Supported Linux Distributions](#) section.

Installation Notes

The Intel compilers use the GlobeTrotter* FLEXlm* electronic licensing technology. A valid license is needed to use the compilers. To install the compilers:

1. Download the compiler package or insert the product CD-ROM.
2. If you downloaded the compiler, untar the compiler package.
3. Become the root user, needed to run rpm, and execute the install script, **"install"**. It's possible to install without root access by unpacking the RPM files with rpm2cpio and editing the compiler environment and configuration files, described below, but this installation method is not supported.
4. Select the compiler or debugger to install. The default RPM options **"-U --replacefiles"** are recommended to update existing files. The recommended installation directory is `/opt/intel`.
5. After installation, the installed packages are listed. Enter **"x"** to exit.
6. Execute the appropriate script to setup the compiler environment variables:
 - `source <install-dir>/compiler60/ia32/bin/iccvars.sh(.csh);`
 - `source <install-dir>/compiler60/ia64/bin/eccvars.sh(.csh).`
7. Install the FLEXlm license: For electronic download, the license is sent via email with install instructions. For the CD-ROM, read the **README.TXT** file.
8. Run the compiler.

Compiler Environment and Configuration Scripts Information

The compilers use the environment variables and the install script creates script files to correctly set the compiler environment. The names and locations are:

`<install-dir>/compiler60/ia32/bin/iccvars.sh(.csh)` for IA-32 application, and
`<install-dir>/compiler60/ia64/bin/eccvars.sh(.csh)` for Itanium processor-based applications.

The installation creates configuration files, `<install-dir>/compiler60/ia32/bin/icc.cfg` and `<install-dir>/compiler60/ia64/bin/ecc.cfg`. They contain common settings and can be used

to add additional default options. Note, if you install a compiler update package, you need to rename the config files to avoid being overwritten.

Installation Warning for RPM 4.0.2

RPM 4.0.2 cannot install to a non-default directory. This has been resolved in RPM 4.0.3.

Uninstalling the Compiler

Become the root user, needed to run rpm, and run the uninstall script::

- `<install-dir>/compiler60/ia32/bin/uninstall` on IA-32 based systems.
- `<install-dir>/compiler60/ia64/bin/uninstall` on Itanium based systems.

Building "Hello World"

Building the classic "Hello World" program is described. After successful installation with a valid FLEXlm license, the configuration scripts, `iccvars.sh(csh)` or `eccvars.sh(csh)`, must be [sourced as described above](#). Then, execute these steps:

1. Create a simple "Hello World" C++ program in a text editor 'hello.C':

```
#include <iostream>
int main() {
    std::cout << "Hello World !" << std::endl;
    return 0; }
```
2. Set up Intel C++ Compiler environment variables as discussed above:
iccvars.sh for IA-32 applications or **eccvars.sh** for Itanium based applications.
3. Compile hello.C :
 - **icc hello.C -o hello**. Creates IA-32 application.
 - **ecc hello.C -o hello**. Creates Itanium-based application
4. Run the executable: **./hello**, it should display "Hello World!".

Getting Started with Compiler Optimizations

The Intel C++ Compiler enables programmers to take full advantage of the advanced performance enhancement features of Intel's latest IA-32 and Itanium processors and includes advanced optimizations. These optimizations are intended for use in product-release builds of applications, not necessarily for earlier phases of application development cycles. In general, increasing the degree of optimization done by the compiler leads to an increase in compile-time and reduced debugging capability. This section describes an optimization methodology with the Intel C++ Compiler.

During the application development, the `"-g -o0"` switches are recommended to allow fast compile times and full debugging with no optimization. To start to optimize, the default optimization `"-o2"` is recommended. The `"-o3"` that enables advanced optimizations. Interprocedural optimization allows the compiler to optimize across different compilation units and can have large performance improvements. Profile guided optimization uses information from running an instrumented executable that allows the compiler to rebuild the application knowing where the majority of the computations are. Of course, not all optimizations are beneficial for all applications.

For additional details on optimizing, the paper, “**Optimizing Applications with the Intel C++ and Fortran Compilers**” is available at

<http://www.intel.com/software/products/compilers/c60/linux>. For complete information on the individual optimizations, please refer to Intel C++ Compiler's Compiler User's Guide at <install-dir>/compiler60/docs/c_ug_lnx.pdf.

Remember to always measure the performance of your application after each optimization added to verify the benefits. The VTune™ Performance Analyzer can be a great help for measuring the performance benefits of each, as well as giving advice on further tuning opportunities, additional information is available at <http://www.intel.com/software/products/vtune/>.

Additional Information

Your feedback is very important to us. To receive technical support for the tools provided in this product and technical information including FAQ's and compiler updates, you need to be registered for an Intel® Premier Support account on our secure web site, <https://premier.intel.com/>. You can register for an Intel Premier Support at <http://support.intel.com/support/performancetools/C/v6>. Note, if you already have access to Intel Premier Support and the “Intel(R) C++ Compiler, Linux” product you do not need to re-register. Compiler support information, including top technical issues and known issues is available at <http://support.intel.com/support/performancetools/C>.

The product release notes contain additional information is located at <install-dir>/compiler60/docs/C++ReleaseNotes.htm. The Compiler User's Guide is located at <install-dir>/compiler60/docs/c_ug_lnx.pdf

Copyright and Legal Information

Intel, Itanium, Pentium, and VTune are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries

* Other names and brands may be claimed as the property of others
Copyright (C) 2001, 2002 Intel Corporation. All Rights Reserved.